What is claimed is:

1. A method of preparing a software patch for transmission to a digital product through a communication link, comprising the steps of:

determining a character set of text characters;

transforming software update data in said software patch into a plurality of text characters of said character set based on a predetermined algorithm.

2. The method of claim 1, wherein said step of determining character set is based on at least one of the following methods:

based on a selection method to select text characters for said character set so that every selected text character in said character set can be correctly transmitted through said communication link to said digital product without any changes.

based on a selection method to select text characters for said character set so that transmitting any selected text characters in said character set does not affect transmitting other selected text characters in said character set through said communication link to said digital product.

3. The method of claim 1, wherein said step of determining character set is to select at least one text character with a value ranged from 32 to 127 for said character set.

4. The method of claim 1, wherein said step of determining character set is not to select at least one of the following text characters,

'@' (At mark)

'_' (Underline Mark)

5. The method of claim 1, wherein said predetermined algorithm is an algorithm to ensure that after said software patch is transformed into said plurality of text characters by said predetermined algorithm, said software patch can be restored without any change or data loss from said plurality of text characters by a second algorithm.

6. The method of claim 1, further comprising:

transmitting said text characters to said digital product through said communication link.

7. The method of claim 6, further comprising:

receiving said text characters by said digital product;

restoring said software patch from said text characters based on a second predetermined algorithm.

8. A method of preparing an embedded software operative in a digital product for field modification by a software patch, comprising the steps of:

determining a plurality of insertion locations for said embedded software based on a predetermined criteria, each of said one insertion locations defining one section of said embedded software;

allocating a plurality of header areas at said plurality of insertion locations, each of said header areas containing update dependent data for a corresponding section;

modifying offset of at least one function call from a section of said embedded software to a second function outside of said section;

directing said function call to a location in one of said header areas;

loading said embedded software with said plurality of header areas into said digital product.

9. The method of claim 8, wherein said update dependent data in a header area of a corresponding section contains at least one call jump to a second function outside of said corresponding section.

10. The method of claim 8, wherein at least one of said sections in said embedded software is inserted with an update processing routine.

11. A method of preparing a digital processor of CPU/DSP for supporting updating embedded software in a digital product, comprising the steps of:

a) preparing a plurality of memory units in said digital processor for controlling software update;

b) preparing a logic circuit in said digital processor to direct execution of said digital processor to a location defined in one memory unit of said plurality of memory units, if execution of said digital processor reaches a second location defined in a second memory unit of said plurality of memory units;

12. The method of claim 11, wherein said plurality of memory units are a plurality of registers of said digital processor.

13. The method of claim 11, wherein said plurality of memory units are a plurality of memory buffers of said digital processor.

14. A method of preparing a digital processor of CPU/DSP for supporting updating embedded software in a digital product, comprising the steps of:

a) *preparing at least one instruction code in the instruction set of said digital* processor for software updating;

b) preparing a logic circuit in said digital processor to direct execution of said digital processor to a second location based on at least one parameter in said instruction code, if execution of said digital processor reaches said instruction code;

15. The method of claim 14, wherein said instruction code contains at least one parameter bit that can be changed from one to zero.